

## **AMENDMENTS TO THE SPECIFICATION: A, B, C**

### **A. Please replace the last paragraph on page 8 with the following paragraph:**

#### **Exemplary Computer System**

Fig. 3 is a block diagram of an exemplary computer system ~~300~~ 302 that includes an inverse query engine ~~302-324~~. The computer system ~~300~~ 302 also includes a

### **B. Please replace the paragraphs on page 9 with the following paragraphs:**

processor 304, output means 306 and input means 308 that allow the computer system ~~300~~ 302 to receive data (e.g. from a mouse, keyboard, etc.) and to send data (e.g. to a printer, etc.). The computer system ~~300~~ 302 also includes a mass storage device 310 (e.g. a hard disk drive, etc.), a network interface 312 (e.g. a network card, modem, etc.) and other miscellaneous hardware 314 typically required for a computer system to function.

The computer system ~~300~~ 302 also includes memory 320, such as Random Access Memory (RAM), in which the inverse query engine ~~302-324~~ is stored. The memory 320 also stores an operating system 322 and other miscellaneous software ~~324~~ 208 that may be required for the computer system ~~300~~ 302 to function properly.

The inverse query engine ~~302-324~~ includes a control module ~~330~~ 336 and a cache ~~332~~ 328 integrated therewith. The cache ~~332~~ 328 stores a filter table ~~334~~ 330 and miscellaneous module ~~336~~ 332 that includes several program, routines or sub-modules necessary for implementation of the systems and methods described herein. Although the cache ~~334~~ 328 is shown stored in RAM 320, it is noted that the cache may be stored in

any practical memory location, such as in Read Only Memory (ROM) (not shown) or on the mass storage device 310. An inverse query engine and its components are discussed in greater detail below.

It is noted that although the inverse query engine ~~302-324~~ is shown as being the only inverse query engine in the computer system ~~300 302~~, it is noted that the inverse query engine 302 could be integrated within a discrete service within the computer system ~~300 302~~. In such an instance, another discrete service having its own inverse query engine could be maintained on the computer system ~~300 302~~. Any practical number of inverse query engines could be present within the computer system ~~300 302~~.

**C. Please replace the paragraphs on page 10 with the following paragraphs:**

**Exemplary Inverse Query Engine**

**Fig. 4** is a block diagram of an exemplary inverse query engine ~~400 324~~ similar to that shown in Fig. 3 and discussed above. The inverse query engine ~~400 324~~ includes a control module ~~402 326~~ and a cache ~~404 328~~. The control module ~~402 326~~ includes an add filter module ~~406 400~~, a remove filter module ~~408 402~~, a matcher ~~410 404~~ and a maintainer ~~412 406~~.

The cache ~~404 328~~ stores a filter table ~~420 330~~ that includes multiple filters ~~422 408~~, a least recently used list ~~424 412~~ that identifies when multiple filters ~~426 408~~ were last used, and an expiration list ~~428 208~~ that identifies expiration times of multiple filters ~~428 408~~.

The add filter module ~~406 400~~ controls functions necessary to receive and add a filter ~~422 408~~ to the filter table ~~420 330~~. The remove filter module ~~408 402~~ controls

functions necessary to remove a filter 422 408 from the filter table. The matcher 410 404 processes messages received by the inverse query engine 400 324 to determine if the messages satisfy any filters 422 408 stored in the filter table 424 330.

The maintainer 412 406 controls cache/filter table maintenance, i.e. the size of the filter table 420 330 using, inter alia, the most recently used list 424 412 and the expiration list 428 208. As will be discussed in greater detail below, the maintainer 412 406 is configured to expire the filter table 420 330 by removing one or more filters 422 408 that have expired.

The maintainer 412 406 is also configured to trim the filter table 420 330 - hence, the cache 404 328 - by determining when the cache 404 328 has grown to a specified maximum size or capacity. The size of the cache 404 328 may be indicated by a size of the filter table 420 330, by the cache 404 328 usage, or by any other method known in the art. When such a determination is made, the maintainer 412 406 is configured to remove one or more filters 422 408 until the cache 404 328 is reduced to an optimal size.